

Информатика



Софронова Наталия Викторовна
*Доктор педагогических наук, профессор,
 г. Чебоксары.*



Григорьев Юрий Владиславович
*Кандидат педагогических наук, доцент,
 г. Чебоксары.*

ИНФОЗНАЙКА-ПРОФИ 2019: примеры задач с решениями

Олимпиадное движение все больше набирает оборотов, особенно в областях, связанных с информационными и коммуникационными технологиями (ИКТ). Общественная организация дополнительного профессионального образования «Чувашское региональное отделение Академии информатизации образования» с 2015 года проводит олимпиаду «Инфознайка-Профи» по информатике и ИКТ для учащихся старших классов [1]. Сайт олимпиады profi.infoznaika.ru. В 2018 году олимпиада была включена в перечень всероссийских школьных олимпиад.

Олимпиада включает два тура: I тур – дистанционный и II тур – очный. Дистанционный тур проводится по программированию. В 2019 году в дистанционном туре приняло участие 606 школьников, учащихся 9-11 классов. Победители I тура 211 человек были приглашены к участию во II туре. В прошлые годы, когда мы проводили очный тур, приезжали 20-30 человек. Чтобы предоставить школьникам возможность попробовать свои силы и во II туре, мы в этом году провели его, используя мобильное приложение, которое можно было скачать с нашего сайта. Мобильное приложение было специально разработано для проведения олимпиады.

Рассмотрим первые пять задач I тура.

1. Двудольный граф

Ограничение времени: 1 секунда

Ограничение памяти: 64Мб

Ввод: стандартный ввод или `input.txt`

Вывод: стандартный вывод или `output.txt`

Граф называется двудольным, если его вершины можно раскрасить в два цвета так, что нет ребер, соединяющих вершины одинакового цвета (то есть каждое ребро идет из вершины 1-го цвета в вершину 2-го)

Дан граф. Требуется проверить, является ли он двудольным, и если да, то раскрасить его вершины.

Входные данные

Во входном файле задано сначала число N – количество вершин графа (не превышает 100). Далее идет матрица смежности – матрица размером $N \times N$ из 0 и 1 (0 обозначает отсутствие ребра, 1 – наличие). Граф неориентированный, без петель.

Выходные данные

В первую строку выведите одно из сообщений YES или NO (в зависимости от того, является ли граф двудольным или нет). В случае ответа YES во второй строке выведите N чисел – цвета, в которые нужно раскрасить вершины.

Примеры

1) Вход 6 0 Выход YES

2) Вх_од 6 0 0 0 0 1 1 0 0 1 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 Вых_од YES

1 1 2 1 2 2 1 1 1 1 1 1

Формат ввода	Формат вывода
3	NO
0 1 1	
1 0 1	
1 1 0	

Решение

```

const nmax=100;
var n:integer;
      a:array[1..nmax,1..nmax] of integer;
      b:array[1..nmax] of byte;
      q:integer;
procedure rec(w,e:integer);
  var i:integer;
  begin
    b[w]:=e;
    for i:=1 to n do
      if a[w,i]=1 then
        if b[i]=0 then rec(i,3-e)

```

```

        else if b[i]=e then begin
            writeln('NO');
            halt;
            end;
    end;
procedure readdata;
var i,j:integer;
begin
    read(n);
    for i:=1 to n do
        for j:=1 to n do
            read(a[i,j]);
        end;
    begin
        readdata;
        for q:=1 to n do
            if b[q]=0 then rec(q,1);
        writeln('YES');
        for q:=1 to n do
            write(b[q], ' ');
        end.
    end.
```

2. Компоненты связности

Ограничение времени: 1 секунда
Ограничение памяти: 64Mb
Ввод: стандартный ввод или input.txt
Вывод: стандартный вывод или output.txt

В неориентированном графе посчитать количество компонент связности. В графе могут быть петли и кратные ребра.

Входные данные

Во входном файле INPUT.TXT записаны сначала два числа N и M, задающие соответственно количество вершин и количество ребер (1<=N<=100, 0<=M<=10000), а затем перечисляются ребра. Каждое ребро задается номерами вершин, которые оно соединяет.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – количество компонент связности.

Формат ввода	Формат вывода
3 4 1 1 1 2 1 3 2 3	1

Решение

```
var n:integer;
    m:longint;
    a:array[1..100,1..100] of byte;
    b:array[1..100] of byte;
    i,j,k:integer;
    an:integer;

procedure rec(v:integer);
    var i:integer;
    begin
        b[v]:=1;
        for i:=1 to n do
            if (b[i]=0) and (a[v,i]=1) then rec(i);
        end;

begin
    read(n,m);
    for k:=1 to m do begin
        read(i,j);
        a[i,j]:=1;
        a[j,i]:=1;
    end;
    an:=0;
    for k:=1 to n do
        if b[k]=0 then begin
            inc(an);
            rec(k);
        end;
    writeln(an);
end.
```

3. Треугольник

Ограничение времени: 1 секунда

Ограничение памяти: 64Mb

Ввод: стандартный ввод или input.txt

Вывод: стандартный вывод или output.txt

Васе на дом задали задачу по геометрии, по трем сторонам треугольника вычислить его периметр и площадь. Но он очень не любил геометрию и попросил вас решить эту задачу.

Входные данные

Во входном файле записано 3 натуральных числа через пробел, не превосходящих 1000 – длины сторон треугольника.

Выходные данные

В первую строку выходного файла выведите число -1 , если треугольника с данными сторонами не существует (Вася неправильно списал условие задачи :), иначе через пробел выведите периметр и площадь данного треугольника. Площадь треугольника выводить с точностью до 5 знаков после запятой (точки).

Формат ввода	Формат вывода
3 4 5	12 6.00000

Решение

```
var p:extended;
    a,b,c:longint;

begin
    read(a,b,c);
    if (a+b<c) or (a+c<b) or (b+c<a) then writeln(-1) else
    begin
        p:=(a+b+c)/2;
        writeln(a+b+c, ' ', sqrt(p*(p-a)*(p-b)*(p-c)):0:5);
    end;
end.
```

4. Двочные последовательности

Ограничение времени: 1 секунда

Ограничение памяти: 64Mb

Ввод: стандартный ввод или *input.txt*

Вывод: стандартный вывод или *output.txt*

Вводится число N . Сгенерируйте в лексикографическом порядке все последовательности длины N , состоящие из чисел 2, 4, 5, в которых количество двоек не превосходит 2-х.

В «лексикографическом порядке» обозначает, что если на первых X местах две последовательности совпадают, а на месте $X+1$ – различаются, то раньше должна идти та из них, в которой число на месте $X+1$ меньше.

$1 \leq N \leq 10$

Формат ввода	Формат вывода
3	2 2 4 2 2 5 2 4 2 2 4 4 2 4 5

	2 5 2
	2 5 4
	2 5 5
	4 2 2
	4 2 4
	4 2 5
	4 4 2
	4 4 4
	4 4 5
	4 5 2
	4 5 4
	4 5 5
	5 2 2
	5 2 4
	5 2 5
	5 4 2
	5 4 4
	5 4 5
	5 5 2
	5 5 4
	5 5 5

Решение

```
const nmax=10;
```

```
var n:integer;  
    a:array[1..nmax] of integer;
```

```
procedure gen(q,w:integer);  
    var i:integer;  
    begin  
        if q=n+1 then begin  
            for i:=1 to n do write(a[i], ' ');  
            writeln;  
            exit;  
        end;  
        if w>0 then begin  
            a[q]:=2;  
            gen(q+1,w-1);  
        end;  
        for i:=4 to 5 do begin  
            a[q]:=i;  
            gen(q+1,w);  
        end;  
    end;
```

```
begin  
  read(n);  
  gen(1,2);  
end.
```

5. Шаблон и слово

Ограничение времени: 1 секунда

Ограничение памяти: 64Mb

Ввод: стандартный ввод или input.txt

Вывод: стандартный вывод или output.txt

Требуется определить, подходит ли заданное слово под заданный шаблон. Шаблон задается большими латинскими буквами, знаками "?" – любой символ, "*" – любая последовательность символов (даже пустая).

Входные данные

В первых двух строках записаны шаблон и слово: в одной строки записан шаблон – последовательность больших латинских букв, "?" и "*", в другой – слово, состоящее только из больших латинских букв (строки короче 100 символов).

Выходные данные

Вывести YES, если слово подходит, или NO, если нет.

Формат ввода	Формат вывода
ABBCDA A*CDA	YES

Решение

```
const  
  maxl = 260;  
var  
  i,j,l1,l2:integer;  
  s1,s2,s3:string;  
  may1,may2:array[0..maxl]of boolean;  
  
function word(s:string):boolean;  
var  
  i:integer;  
begin  
  word := false;  
  for i:=1 to length(s) do  
    if not (s[i] in ['A'..'Z']) then exit;  
  word := true;  
end;
```

```
begin
  readln(s1);
  readln(s2);
  if word(s2) then
    begin
      s3 := s1;
      s1 := s2;
      s2 := s3;
    end;
  l1:= length(s1); { word }
  l2:= length(s2);
  for j:=0 to l1 do
    may1[j] := j=0;
  for i:=1 to l2 do
    begin
      may2[0] := may1[0] and ( s2[i] = '*' );
      for j:=1 to l1 do
        begin
          if s2[i] = '?' then may2[j] := may1[j-1]
          else
            if s2[i] = '*' then may2[j] := may1[j] or may1[j-1]
or may2[j-1]
          else
            may2[j] := may1[j-1] and (s1[j] = s2[i]);
          end;
        may1 := may2;
      end;
    if may1[l1] then writeln('YES')
    else writeln('NO');
  end.
```

Обсуждение задач состоялось после проведения олимпиады в режиме вебинара. Вебинар можно посмотреть по ссылке

<https://www.youtube.com/watch?v=CwXe0OYvWuk&t=36s>.

Литература:

1. Софронова, Н. В. Образовательная деятельность общественной организации: возможности и проблемы // Педагогическая информатика. – № 3. – 2018. – С. 171-176.